This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1.   (Currently Amended)   Method for improving a processing time of received data in packet oriented applications in a data transmission of data flows between a transmitter and a receiver, each transmission involving a first and a second protocol layer and being carried via a communication network, wherein said method comprises:

providing a data flow at the first protocol layer as data packets of the first protocol layer;

releasing data from the first protocol layer to the second protocol layer in the transmitter;

dividing the data of the first protocol layer into consecutive data packets of the second protocol layer by generating a sequence of data packets with sequence numbers, wherein a data packet of the second protocol layer contains data from only one data packet of the first protocol layer;

transmitting the data packets of the second protocol layer to the receiver via the communication network;

sorting received data packets of the second protocol layer at the receiver according to the sequence of the data packets;

allocating received data packets of the second protocol layer to data packets of the first protocol layer, in the second protocol layer; and

upon a data packet of the first protocol layer being completely generated from a group of data packets of the second protocol layer allocated to the first protocol layer, examining said completely generated data packet for an association to a data flow, and releasing said completely generated data packet to the first protocol layer without considering the non-delivery of prior data packets of said second protocol layer not a part of said completely generated data packet of said first protocol layer.

2.   (Cancelled)

3.   (Previously Presented)   Method according to claim 1 wherein the data packets of the second protocol layer are numbered consecutively and marked by corresponding sequence numbers.

4.   (Previously Presented)   Method according to claim 1, wherein the first protocol layer supports at least two transmission modes, a reliable and an unreliable mode.

5.   (Original)   Method according to claim 4, wherein the data packets of the second protocol layer are corrected by means of repeated transmission in case of a transmission error and by using the reliable transmission mode.

6.   (Previously Presented)   Method according to claim 1, wherein the data packets of the first protocol layer are clearly differentiated from each other by means of separators.

7.   (Original)   Method according to claim 3, wherein the received data packets are sorted into a sequence corresponding to a sequence number.

8.   (Previously Presented)   Method according to claim 3, wherein each sequence number is an RLP (Radio Link Protocol) sequence number or an RLC (Radio Link Control) sequence number.

9.   (Previously Presented)   Method according to claim 1, wherein the received data packets are stored in a buffer of the receiver.

10.   (Previously Presented)   Method according to claim 1, wherein a data packet of the first protocol layer is given a status of a completely generated data packet, if both an initial and an end mark within data packets of the second protocol layer have

correctly been received, and if all data packets of the second protocol layer lying in between have correctly been received.

11.   (Previously Presented)   Method according to claim 1, wherein the completely generated data packets of the first protocol layer are examined according to the rules of an encapsulating process, for identifying packets of additional protocol layers.

12.   (Previously Presented)   Method according to claim 1, wherein at least one control field comprising control data is provided in the completely generated data packets of the first protocol layer, for delivering the information of a pertinent data flow.

13.   (Original)   Method according to claim 12, wherein the control data are appended to the actual data sequences as control fields in the corresponding protocol layers in form of a header and/or a tail.

14.   (Previously Presented)   Method according to claim 1, wherein a data flow is differentiated by means of certain control data in the control fields provided therefor.

15.   (Original)   Method according to claim 14, wherein the control data for differentiating data flows are the addresses of the transmitter and/or receiver in form of source addresses, destination addresses and port numbers.

16.   (Previously Presented)   Method according to claim 1, wherein the data packets of the first protocol layer are directly released to the first protocol layer on the second protocol layer, if the data packets of the second protocol layer have firstly been received completely and correctly, and if secondly it has been guaranteed that the data possibly buffered by the receiver of the second protocol layer do not contain additional

data packets of the first protocol layer belonging to the same data flow of the data packets of the first protocol layer to be released.

17.   (Previously Presented)   Method according to claim 1, wherein on the second protocol layer the data packets of the first protocol layer are directly released to the first protocol layer, if said data packets have been received completely and correctly.

18.   (Original)   Method according to claim 1, wherein the data packets of the first protocol layer are IP datagrams and the data packets of the second protocol layer are PPP frames, wherein PPP frames are corrected by means of repeated transmission when an error occurs.

19.   (Original)   Method according to claim 1, wherein the data packets of the first protocol layer are PPP frames and the data packets of the second protocol layer are RLP data frames.

20.   (Original)   Method according to claim 1, wherein the data transmission is performed via an IP network and a mobile communication network.

21.   (Original)   Method according to claim 1, wherein the packet oriented applications are internet applications.

22.   (Previously Presented)   Method according to claim 21, wherein an internet application is transmitted by means of the transport protocol Transmission Control Protocol (TCP).

23.   (Previously Presented)   Method according to claim 21, wherein an internet application is transmitted by means of the transport protocol User Datagram Protocol (UDP).

24.   (Currently Amended)   Device for improving a processing time of received data in packet oriented applications in a data transmission of data flows between a transmitter and a receiver, each transmission involving a first and a second protocol layer and being carried via a communication network, said device comprising:

means for providing data packets of a first protocol layer to a second protocol layer, wherein the data of the first protocol layer is divided into consecutive data packets of the second protocol layer by generating a sequence of data packets with sequence numbers, and wherein a data packet of the second protocol layer contains data from only one data packet of the first protocol layer;

transmitting means for transmitting the data packets of the second protocol layer;

receiving means for receiving the transmitted data packets;

means for sorting the received data packets according to the sequence of the data packets, and for allocating the received data packets to data packets of the first protocol layer, in the second protocol layer;

recognizing means for recognizing that a data packet of the first protocol layer has been completely generated from a group of data packets of the second protocol layer allocated to the first protocol layer;

means for examining said completely generated data packet for an association to a data flow; and

releasing means for releasing said completely generated data packet to the first protocol layer without considering the non-delivery of prior data packets of said second protocol layer not a part of said completely generated data packet of said first protocol layer.

25.   (Original)   Device according to claim 24 comprising a buffer for temporarily storing the received data packets of the second protocol layer.